

IN THE  
UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Venkatesh Krishnan et al.

Confirmation No.: 8205

Application No.: 09/350,492

Examiner: Tang K.

Filing Date: 7-9-99

Group Art Unit: 2195

Title: TWO TIER ARRANGEMENT FOR THREADS SUPPORT IN A VIRTUAL MACHINE

Mail Stop Appeal Brief-Patents  
Commissioner For Patents  
PO Box 1450  
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL BRIEF

Sir:

Transmitted herewith is the Appeal Brief in this application with respect to the Notice of Appeal filed on 7-21-05.

The fee for filing this Appeal Brief is (37 CFR 1.17(c)) \$500.00.

(complete (a) or (b) as applicable)

The proceedings herein are for a patent application and the provisions of 37 CFR 1.136(a) apply.

( ) (a) Applicant petitions for an extension of time under 37 CFR 1.136 (fees: 37 CFR 1.17(a)-(d) for the total number of months checked below:

( ) one month	\$120.00
( ) two months	\$450.00
( ) three months	\$1020.00
( ) four months	\$1590.00

( ) The extension fee has already been filled in this application.

(X) (b) Applicant believes that no extension of time is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition and fee for extension of time.

Please charge to Deposit Account **08-2025** the sum of \$500.00. At any time during the pendency of this application, please charge any fees required or credit any over payment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.16 through 1.21 inclusive, and any other sections in Title 37 of the Code of Federal Regulations that may regulate fees. A duplicate copy of this sheet is enclosed.

(X) I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, Alexandria, VA 22313-1450. Date of Deposit: 9-21-05

OR

( ) I hereby certify that this paper is being transmitted to the Patent and Trademark Office facsimile number \_\_\_\_\_ on \_\_\_\_\_

Number of pages:

Typed Name: Paul H. Horstmann

Signature:

Respectfully submitted,

Venkatesh Krishnan et al.

By

Paul H. Horstmann

Attorney/Agent for Applicant(s)  
Reg. No. 36,167

Date: 9-21-05

Telephone No.: (310) 376-0218



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re Application of:

Venkatesh Krishnan et al. )  
Application No: 09/350,492 ) Examiner: Tang K.  
Filed: 7-9-99 ) Art Unit: 2195  
For: TWO TIER ARRANGEMENT FOR )  
THREADS SUPPORT IN A )  
VIRTUAL MACHINE )  
I hereby certify that this correspondence is being  
deposited with the United States Postal Service  
as first class mail with sufficient postage in an  
envelope addressed to the Commissioner for  
Patents, P.O. Box 1450, Alexandria, VA 22313-  
1450 on 9-21-05

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Date of Deposit

Paul H. Horstmann  
Name of Person Mailing Correspondence

Paul H. Horstmann 9-21-05  
Signature Date

**Appellant's Brief (Pursuant to 37 C.F.R. §41.37)**

Dear Sir:

Applicant/Appellant submits this Appeal Brief in connection with the above-referenced patent application which is on appeal to the Board of Patent Appeals and Interferences.

09/26/2005 MBIZUNES 00000057 082025 09350492

01 FC:1402 500.00 DA



## TABLE OF CONTENTS

<b>REAL PARTY IN INTEREST .....</b>	<b>3</b>
<b>RELATED APPEALS AND INTERFERENCES .....</b>	<b>3</b>
<b>STATUS OF THE CLAIMS .....</b>	<b>3</b>
<b>STATUS OF AMENDMENTS .....</b>	<b>3</b>
<b>SUMMARY OF CLAIMED SUBJECT MATTER.....</b>	<b>4</b>
<b>GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL .....</b>	<b>5</b>
I: REJECTION OF CLAIMS 30-35 AND 47-52 AS BEING ANTICIPATED BY NELSON. ....	5
II: REJECTION OF CLAIMS 36 AND 38-42 AS BEING OBVIOUS IN VIEW OF NELSON AND DELAGI. ....	5
III: REJECTION OF CLAIM 37 AS BEING OBVIOUS IN VIEW OF NELSON AND BUCHER....	5
IV: REJECTION OF CLAIMS 43-46 AS BEING OBVIOUS IN VIEW OF NELSON AND FARRELL. ....	5
<b>ARGUMENT.....</b>	<b>6</b>
I: CLAIMS 30-35 AND 47-52 ARE NOT ANTICIPATED BY NELSON BECAUSE NELSON DOES NOT DISCLOSE THE LIMITATIONS OF INDEPENDENT CLAIMS 30 AND 47. ....	6
<i>A. Nelson does not disclose threads support for a set of threads of an application program that executes on a virtual machine as claimed in claims 30 and 47.....</i>	6
<i>B. Nelson does not disclose a virtual machine having a threads interface layer that provides a standard threads interface for a set of threads associated with an application program executed by the virtual machine such that the standard threads interface does not depend on an underlying platform of the virtual machine as claimed in claims 30 and 47. ....</i>	7
<i>C. Nelson does not disclose a virtual machine having a native threads interface layer that adapts a threads interface layer in a virtual machine to an underlying platform of the virtual machine as claimed in claims 30 and 47.....</i>	8
II: CLAIMS 36 AND 38-42 ARE NOT OBVIOUS IN VIEW OF NELSON AND DELAGI BECAUSE NELSON AND DELAGI DO NOT DISCLOSE OR SUGGEST THE LIMITATIONS OF CLAIM 30.....	9
III: CLAIM 37 IS NOT OBVIOUS IN VIEW OF NELSON AND BUCHER BECAUSE NELSON AND BUCHER DO NOT DISCLOSE OR SUGGEST THE LIMITATIONS OF CLAIM 30.....	10
IV: CLAIMS 43-46 ARE NOT OBVIOUS IN VIEW OF NELSON AND FARRELL BECAUSE NELSON AND FARRELL DO NOT DISCLOSE OR SUGGEST THE LIMITATIONS OF CLAIM 30. ....	11
<b>CONCLUSION .....</b>	<b>12</b>
<b>CLAIMS APPENDIX.....</b>	<b>13</b>
<b>EVIDENCE APPENDIX.....</b>	<b>16</b>
<b>RELATED PROCEEDINGS APPENDIX .....</b>	<b>17</b>

### **REAL PARTY IN INTEREST**

The real party in interest in this application is Hewlett-Packard Development Company, L.P.

### **RELATED APPEALS AND INTERFERENCES**

Appellant is unaware of any other related appeals or interferences that may directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

### **STATUS OF THE CLAIMS**

Claims 30-35 and 47-52 stand rejected under 35 U.S.C. §102(e) as being unpatentable over U.S. Patent Application Publication US 2001/0049686 of *Nelson et al.* ("*Nelson*").

Claims 36 and 38-42 stand rejected under 35 U.S.C. §103(a) as being unpatentable over *Nelson* and U.S. Patent no. 3,858,182 of *Delagi et al.* ("*Delagi*").

Claim 37 stands rejected under 35 U.S.C. §103(a) as being unpatentable over *Nelson* and U.S. Patent no. 5,421,014 of *Bucher* ("*Bucher*").

Claims 43-46 stand rejected under 35 U.S.C. §103(a) as being unpatentable over *Nelson* and U.S. Patent no. 5,630,128 of *Farrell et al.* ("*Farrell*").

Appellant appeals the rejection of all of the pending claims 30-52. Claims 30-52 as currently pending are set forth in the attached Appendix.

### **STATUS OF AMENDMENTS**

Appellant is unaware of any amendments filed after the Final Office Action mailed April 21, 2005 which finally rejected claims 30-52.

## **SUMMARY OF CLAIMED SUBJECT MATTER**

Independent claims 30 and 47 are directed to a two tier<sup>1</sup> arrangement for providing threads support to an application program that executes on a virtual machine. A two tier arrangement for threads support according to claims 30 and 47 enhances the adaptability of a virtual machine to different underlying operating system and hardware platforms<sup>2</sup>. A two tier arrangement for adapting threads support in a virtual machine according to claims 30 and 47 includes a threads interface layer and a native threads interface layer<sup>3</sup>. The threads interface layer provides a standard threads interface for a set of threads associated with an application program executed by a virtual machine such that the standard threads interface does not depend on an underlying platform of the virtual machine while the native threads interface layer adapts the threads interface layer to the underlying platform<sup>4</sup>.

---

<sup>1</sup> The two tier arrangement is recited in claims 30-52 as a “threads interface layer” that provides a standard threads interface to an application program running on a virtual machine and a “native threads interface layer” for adapting the threads interface layer to an underlying platform. Appellant will use the term “two tier arrangement” for threads support as a shorthand way of referring to the two threads interface layers recited in claims 30-52.

<sup>2</sup> See Appellant’s Specification on page 4, lines 3-6.

<sup>3</sup> See Appellant’s Specification on page 4, lines 6-9, elements 14 and 16 of Figure 1, and elements 100 and 102 of Figure 3.

<sup>4</sup> See Appellant’s Specification on page 4, lines 9-15, elements 12, 18, 20, and 30-32 of Figure 1, and elements 100 and 102 of Figure 3.

**GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL**

- I: Rejection of claims 30-35 and 47-52 as being anticipated by *Nelson*.**
- II: Rejection of claims 36 and 38-42 as being obvious in view of *Nelson* and *Delagi*.**
- III: Rejection of claim 37 as being obvious in view of *Nelson* and *Bucher*.**
- IV: Rejection of claims 43-46 as being obvious in view of *Nelson* and *Farrell*.**

## ARGUMENT

### **I: Claims 30-35 and 47-52 are not anticipated by *Nelson* because *Nelson* does not disclose the limitations of independent claims 30 and 47.**

Appellant respectfully submits that claims 30 and 47, and claims 31-35 and 48-52 which depend from claims 30 and 47, are not anticipated by *Nelson* because *Nelson* does not disclose threads support for a set of threads of an application program that executes on a virtual machine as claimed in claims 30 and 47. Moreover, *Nelson* does not disclose a threads interface layer (TIL) that provides a standard threads interface for a set of threads associated with an application program executed by a virtual machine as claimed in claims 30 and 47. Furthermore, *Nelson* does not disclose a native threads interface layer (NTIL) that adapts the threads interface layer to an underlying platform as claimed in claims 30 and 47.

#### **A. *Nelson* does not disclose threads support for a set of threads of an application program that executes on a virtual machine as claimed in claims 30 and 47.**

Appellant submits that *Nelson* does not disclose threads support for a set of threads of an application program that executes on a virtual machine as claimed in claims 30 and 47 much less a two tier arrangement for adapting threads support as claimed in claims 30 and 47. Instead, *Nelson* discloses programming tools for developing network management software for Java environments. (*Nelson*, Abstract, lines 1-8).

*Nelson* does disclose a Java virtual machine (the JNI layer 604 of *Nelson* at paragraph 0039, lines 15-18) that includes native Solaris threads (*Nelson*, paragraph 0039, lines 19-23) but nothing in *Nelson* indicates that the native Solaris threads in the Java virtual machine in the JNI layer 604 may be used to support a set of threads of an application program executing on the virtual machine as claimed in claims 30 and 47. Appellant has found only one use of the word “threads” in the entire disclosure of *Nelson* at paragraph 0039, line 19, and that is in connection with threads support in the Java virtual machine

in the JNI layer 604 and not in connection with any application programs that may run on the Java virtual machine in the JNI layer 604.<sup>5</sup>

**B. *Nelson* does not disclose a virtual machine having a threads interface layer that provides a standard threads interface for a set of threads associated with an application program executed by the virtual machine such that the standard threads interface does not depend on an underlying platform of the virtual machine as claimed in claims 30 and 47.**

Appellant submits that *Nelson* does not disclose a virtual machine having a threads interface layer that provides a standard threads interface for a set of threads associated with an application program executed by the virtual machine such that the standard threads interface does not depend on an underlying platform of the virtual machine as claimed in claims 30 and 47. Instead, *Nelson* discloses a virtual machine with threads support that does depend on an underlying platform (Solaris) of the virtual machine (*Nelson*, paragraph 0039, lines 15-23).

For example, *Nelson* discloses a JNI layer 604 that includes A Java Native Interface (“JNI”) Layer 604 includes a library of C and/or C++ methods configured to define a Java Virtual Machine (“JVM”) that provides translation of CMIS to Java. (*Nelson*, paragraph 0039, lines 15-18) (emphasis added) and states that the JNI layer 604

this layer also includes Solaris threads configured to provide additional support when the MPA is running in conjunction with the Solaris operating system (available commercially from Sun Microsystems...). (*Nelson*, paragraph 0039, lines 19-22) (emphasis added).

In response to appellant’s argument that *Nelson* discloses a virtual machine with only thread support that does depend on an underlying platform, the examiner has stated that *Nelson* discloses a Java virtual machine

---

<sup>5</sup> It is submitted that the examiner consistently fails to distinguish between a virtual machine and an application program that runs on the virtual machine. The two tier arrangement for threads support of claims 30-52 enables execution of a set of threads of an application program that runs on a virtual machine whereas the native Solaris threads in the virtual machine of *Nelson* enables the virtual machine itself to have threads.

and that Java is platform independent.<sup>6</sup> (Page 8, paragraph numbered 23, Office Action, 4/21/05). Appellant respectfully submits that a Java virtual machine is not platform-independent as implied by the examiner. It is submitted that although a Java application program that runs on a Java virtual machine may be regarded as platform-independent, the Java virtual machine itself<sup>7</sup> is platform-dependent in that it is adapted to run on a particular underlying operating system and hardware architecture. (See statement of Art Background starting on page 1 of appellant's specification). (See also the teaching in paragraph 39 of *Nelson* of using platform-dependent Solaris threads to implement a Java virtual machine).

**C. *Nelson* does not disclose a virtual machine having a native threads interface layer that adapts a threads interface layer in a virtual machine to an underlying platform of the virtual machine as claimed in claims 30 and 47.**

Appellant submits that *Nelson* does not disclose a virtual machine having a native threads interface layer that adapts a threads interface layer in the virtual machine to an underlying platform of the virtual machine as claimed in claims 30 and 47. This follows from the fact that *Nelson* does not disclose a threads interface layer in a virtual machine that provides a standard threads interface that does not depend on an underlying platform of the virtual machine as claimed in claims 30 and 47.

---

<sup>6</sup> The examiner appears to argue that if a Java virtual machine is platform-independent then the threads support in a Java virtual machine must also be platform-independent even though *Nelson* discloses only platform-dependent (Solaris) threads.

<sup>7</sup> Again, the examiner fails to distinguish between a virtual machine and an application program that runs on the virtual machine.

**II: Claims 36 and 38-42 are not obvious in view of *Nelson* and *Delagi* because *Nelson* and *Delagi* do not disclose or suggest the limitations of claim 30.**

Appellant respectfully submits that claims 36 and 38-42, which depend from claim 30, are not obvious in view of *Nelson* and *Delagi* because *Nelson* and *Delagi* do not disclose or suggest a virtual machine having a two tier arrangement for threads support that includes a threads interface layer that provides a standard threads interface and that further includes a native threads interface layer that adapts the standard threads interface to an underlying platform as claimed in claim 30. Appellant has shown that *Nelson* does not disclose or suggest a virtual machine having threads support as claimed in claim 30. *Delagi* discloses multiprogramming hardware support in an underlying platform of a virtual machine (*Delagi*, col. 4, line 4 through col. 5, line 4) rather than a virtual machine having a two tier arrangement for threads support that includes a threads interface layer that provides a standard threads interface and a native threads interface layer that adapts the standard threads interface to an underlying platform as claimed in claim 30.

**III: Claim 37 is not obvious in view of *Nelson* and *Bucher* because *Nelson* and *Bucher* do not disclose or suggest the limitations of claim 30.**

Appellant respectfully submits that claim 37, which depends from claim 30, is not obvious in view of *Nelson* and *Bucher* because *Nelson* and *Bucher* do not disclose or suggest a virtual machine with a two tier arrangement for threads support as claimed in claim 30. Appellant has shown that *Nelson* does not disclose or suggest a two tier arrangement for threads support as claimed in claim 30. *Bucher* discloses data structures for storing thread context information (*Bucher*, col. 3, lines 29-35) for bus accesses to peripheral devices (*Bucher*, col. 4, line 66 through col. 5, line 47) rather than a two tier arrangement for threads support as claimed in claim 30.

**IV: Claims 43-46 are not obvious in view of *Nelson* and *Farrell* because *Nelson* and *Farrell* do not disclose or suggest the limitations of claim 30.**

Appellant respectfully submits that claims 43-46, which depend from claim 30, are not obvious in view of *Nelson* and *Farrell* because *Nelson* and *Farrell* do not disclose or suggest a virtual machine having a two tier arrangement for threads support as claimed in claim 30. Appellant has shown that *Nelson* does not disclose or suggest a virtual machine having a two tier arrangement for threads support as claimed in claim 30.

*Farrell* does not disclose or suggest a virtual machine having a two tier arrangement for threads support that includes a threads interface layer that provides a standard threads interface to application programs that does not depend on an underlying platform and that further includes a native threads interface layer that adapts the standard threads interface to an underlying platform as claimed in claim 30. Instead, *Farrell* discloses threads support that does depend on underlying platform of a virtual machine (*Farrell*, col. 3, lines 27-49). For example, Figure 1 of *Farrell* shows a set of application-callable threads support routines that are part of an operating system 10, i.e. an underlying platform of a virtual machine. (*Farrell*, col. 3, lines 43-46).

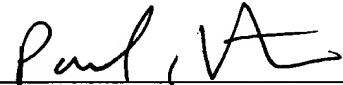
## CONCLUSION

Appellant respectfully submits that the stated rejections cannot be maintained in view of the arguments set forth above. Appellant respectfully submits that all of the claims 30-52 are patentable under 35 U.S.C. §§102,103 over the references cited by the Examiner and requests that the Board of Patent Appeals and Interferences direct allowance of the rejected claims.

Respectfully submitted,

By

Date: 9-21-05

  
\_\_\_\_\_  
Paul H. Horstmann

Reg. No. 36,167

## **CLAIMS APPENDIX**

30. A system for adapting threads support in a virtual machine to an underlying platform of the virtual machine, comprising:

threads interface layer (TIL) that provides a standard threads interface for a set of threads associated with an application program of the virtual machine such that the standard threads interface does not depend on the underlying platform;

native threads interface layer (NTIL) for adapting the TIL to the underlying platform such that a set of routines in the TIL use a set of routines in the NTIL to support the threads.

31. The system of claim 30, wherein the NTIL is adapted to an operating system of the underlying platform.

32. The system of claim 31, wherein the NTIL is adapted to use a set of thread support routines provided by the operating system.

33. The system of claim 31, wherein the NTIL is adapted to use a set of routines provided by the operating system that perform equivalent functions of functions in the native threads interface layer (NTIL).

34. The system of claim 30, wherein the NTIL is adapted to a hardware architecture of the underlying platform.

35. The system of claim 30, wherein the standard threads interface is a Java threads class.

36. The system of claim 30, wherein the routines in the TIL maintain a set of context information for each thread in terms of the virtual machine.

37. The system of claim 30, wherein the routines in the NTIL maintain a set of context information for each thread in terms of the underlying platform.
38. The system of claim 30, wherein the native threads support routines include a routine for suspending a particular thread.
39. The system of claim 30, wherein the native threads support routines include a routine for resuming a particular thread.
40. The system of claim 30, wherein the native threads support routines include a routine for waiting for completion of a particular thread.
41. The system of claim 30, wherein the native threads support routines include a routine for yielding execution to another thread.
42. The system of claim 30, wherein the native threads support routines include a routine for stopping execution of a particular thread and for cleaning up a set of structures associated with the particular thread.
43. The system of claim 30, wherein the native threads support routines include a routine for setting a priority of a particular thread.
44. The system of claim 30, wherein the native threads support routines include a routine for obtaining a priority of a particular thread.
45. The system of claim 30, wherein the native threads support routines include a routine for obtaining an identifier of a currently executing thread.
46. The system of claim 30, wherein the native threads support routines include a routine for selecting a particular thread for execution.

47. A method for adapting threads support in a virtual machine to an underlying platform, comprising:

- providing a threads interface layer (TIL) having a standard threads interface in the virtual machine for a set of threads associated with an application program that executes under the virtual machine such that the standard threads interface does not depend on the underlying platform;
- providing a native threads interface layer (NTIL) for adapting the TIL to the underlying platform such that a set of routines in the TIL use a set of routines in the NTIL to support the threads.

48. The method of claim 47, wherein providing an NTIL includes adapting the NTIL to an operating system of the underlying platform.

49. The method of claim 48, wherein adapting the NTIL to an operating system includes adapting the NTIL to use a set of thread support routines provided by the operating system.

50. The method of claim 48, wherein adapting the NTIL to an operating system includes adapting the NTIL to use a set of routines provided by the operating system that perform equivalent functions of functions in the NTIL.

51. The method of claim 47, wherein providing an NTIL includes adapting the NTIL to a hardware architecture of the underlying platform.

52. The method of claim 47, wherein providing an NTIL having a standard threads interface includes providing a Java threads class.

**EVIDENCE APPENDIX**

None.

**RELATED PROCEEDINGS APPENDIX**

None.